DISCRETE
APPLIED
MATHEMATICS



# An optimal algorithm for finding the minimum cardinality dominating set on permutation graphs

H.S. Chao[a], F.R. Hsu[b], R.C.T. Lee *

[a] Department of Computer Sciences, National Tsing Hua University, Hsinchu, Taiwan 30043
[b] Department of Accounting, Providence University, Shalu, Taichung Hsien, Taiwan 43301

Received 9 May 1997; accepted 8 September 1998

**Abstract**

Given a permutation graph $G$ with its corresponding permutation $\pi$, we present an algorithm for finding a minimum cardinality dominating set for $G$. Our algorithm runs in $O(n)$ time in amortized sense where $n$ is the number of vertices in $G$. Hence, it is optimal. The best previous result is an $O(n \log \log n)$ algorithm. 

## 1. Introduction

Let $\pi = [\pi_1, \pi_2, \ldots, \pi_n]$ be a permutation of the numbers $1, 2, \ldots, n$. We can construct a graph $G[\pi] = (V, E)$ with vertex set $V = \{1, \ldots, n\}$ and edge set $E$:

$$(i, j) \in E \Leftrightarrow (i - j)(\pi^{-1}(i) - \pi^{-1}(j)) < 0,$$

where $\pi^{-1}(i)$ is the position of $i$ in $\pi = [\pi_1, \pi_2, \ldots, \pi_n]$. An undirected graph $G$ is a *permutation graph* [8] if there is a permutation $\pi$ such that $G$ is isomorphic to $G[\pi]$. In this paper, we assume that the input is a permutation $\pi = [\pi_1, \pi_2, \ldots, \pi_n]$.

A permutation graph is an intersection graph based upon the *permutation diagram* [8], which is defined as follows: Write the numbers $1, 2, \ldots, n$ horizontally from left to right. Under every $i$, write the number $\pi(i)$. Draw line segments connecting $i$ in the top row and $i$ in the bottom row, for each $i$. It is easy to see that two vertices $i$ and $j$ of $G[\pi]$ are adjacent if and only if the corresponding line segments of $i$ and $j$ intersect. Fig. 1 shows the permutation graph $G[\pi]$ and its corresponding permutation diagram of a permutation $\pi[3, 1, 5, 7, 4, 2, 6]$.

For an undirected graph $G = (V, E)$, a vertex $i \in V$ is said to *dominate* another vertex $j \in V$ if $(i, j) \in E$. For any two sets $S_1$ and $S_2$, the set $S_2 \setminus S_1$ is the set of all elements which belong to $S_2$ but do not belong to $S_1$. For an undirected graph $G = (V, E)$,

* Correspondence address: Office of the President, Providence University, Shalu, Taichung Hsien, Taiwan 43301.

*E-mail address:* rctlee@simon.pu.edu.tw (R.C.T. Lee)

0166-218X/00/$ - see front matter 
PII: S0166-218X(00)00145-0

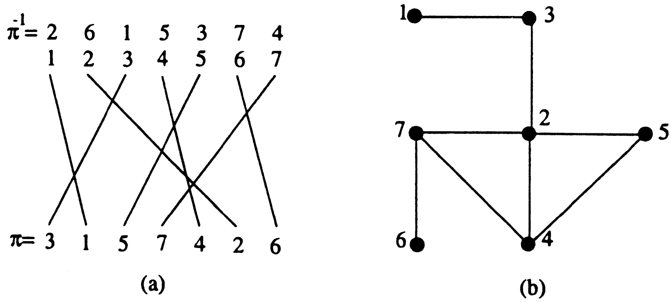


Fig. 1. (a) The permutation diagram. (b) A permutation graph.

a vertex set $S_1$ is said to *dominate* another vertex set $S_2$ if every vertex in $S_2 \backslash S_1$ is dominated by at least one vertex in $S_1$. Let $S_1 \rhd S_2$ and $S_1 \not\rhd S_2$ denote that $S_1$ dominates $S_2$ and $S_1$ does not dominate $S_2$, respectively. A vertex set $S$ is said to be a *dominating set* for $G$ if $S \rhd V$. The *minimum cardinality dominating set* problem (MCDS) and the *minimum weighted dominating set* problem (MWDS) are to find a dominating set $S$ for $G$ where the number and the total weight of vertices of $S$ are minimized, respectively.

Both of the MCDS and the MWDS problems are NP-hard for general graphs [7]. On permutation graphs, Farber and Keil [6] proposed $O(n^2)$ and $O(n^3)$ algorithms for the MCDS and MWDS problems, respectively, based upon the dynamic programming method. Later, by utilizing the monotone ordering among the intermediate terms of the recursive formula in [6], Tsai and Hsu [14] improved the time-complexities to $O(n \log\log n)$ and $O(n^2 \log^2 n)$, respectively. For the MWDS problem, Liang et al. [11] proposed an $O(n(n+m))$ algorithm, where $m$ is the number of edges. Later, they reduced the time-complexity to $O(n+m)$ time [12]. Many related problems on permutation graphs have been studied [5]. For the minimum weight independent dominating set problem. Farber and Keil [6] proposed an $O(n^3)$ algorithm. In [3], an $O(n^2)$ algorithm was presented by Brandstädt and Kratsch. The best-known result is an $O(n \log n)$ algorithm proposed by Atallah and Kosaraju [2]. Colbourn and Stewart [4] presented an $O(n^3)$ algorithm for the minimum weighted connected dominating set problem. This result was improved to $O(m + n \log n)$ by Arvind and Rangan [1], and to $O(m + n)$ by Liang [10]. An optimal $O(n)$ algorithm for the minimum cardinality connected dominating set problem was proposed by Ibarra and Zheng [9].

In this paper, we propose an optimal $O(n)$ algorithm for solving the MCDS problem on permutation graphs. Our algorithm is based on a new recursive formula by using the dynamic programming method, which is different from the formula in [6]. There is also a monotone ordering among the intermediate terms of our recursive formula. Then, according to our recursive formula and the monotone ordering, we propose new updating rules so that we can design an optimal linear time algorithm in amortized sense for the MCDS problem on permutation graphs.

The remainder of this paper is organized as follows: In Section 2, we describe our recursive formula of the dynamic programming. In Section 3, we present the new rules for making use of the monotone ordering. Our linear time algorithm is described in Section 4. Section 5 contains some conclusions.

## 2. The dynamic programming approach

In the following, we shall describe our basic approach based upon the dynamic programming approach. Essentially, we want to find an MCDS of $\{\pi_1, \pi_2, \ldots, \pi_n\}$ dominating $\{1, 2, \ldots, n\}$. We first define $V_i$. For any $i$, $V_i = \{\pi_1, \pi_2, \ldots, \pi_i\}$. In other words, $V_i$ is the set of the first $i$ elements in $\pi$. For instance, for the permutation diagram in Fig. 1(a), if $i = 4$, then $V_4 = \{3, 1, 5, 7\}$. From the top point of view, our approach finds an MCDS of $V_i$ dominating $V_i$, starting from $i = 1$ to $i = n$. Yet, in the process, the result of finding an MCDS of $V_{i-1}$ dominating $V_{i-1}$ can be used to produce an MCDS of $V_i$ dominating $V_i$.

To find an MCDS of $V_i$ dominating $V_i$, we consider whether we select $\pi_i$ or not. If $\pi_i$ is selected, all vertices in $V_i$ greater than $\pi_i$ are dominated by $\pi_i$ and all vertices in $V_i$ smaller than $\pi_i$ are not dominated by $\pi_i$, as can be understood by examining a permutation diagram. Consider Fig. 1(a) again. Suppose that $i = 5$, $V_i = \{3, 1, 5, 7, 4\}$. If vertex 4 is selected, in $V_i$, 5 and 7 are greater than 4. They are dominated by vertex 4. Besides, it can be easily seen that in $V_i$, 3 and 1 are smaller than 4, and they are not dominated by vertex 4. Thus, given an $i$, we are interested in the subset of $V_i$ containing elements smaller than $\pi_i$. In our approach, we use another parameter $j$ and define $V_{i,j}$ to be the subset of $V_i$ containing all elements smaller than or equal to $j$. We start from $j = 1$ to $j = n$. We shall show later that after finding MCDS's of $V_{i-1}$ dominating $V_{i-1,j}$, $j = 1, 2, \ldots, n$, then the MCDS's of $V_i$ dominating $V_{i,j}$, $j = 1, 2, \ldots, n$, can be computed easily. As will become clear later, we shall add another element into $V_i$ to define a new term $V_i^*$. For each $i$, $1 \leqslant i \leqslant n$, $\pi_i^*$ is to be the minimum number over the suffix $\pi_i, \pi_{i+1}, \ldots, \pi_n$. Then $V_i^* = V_i \cup \{\pi_i^*\}$. The significance of $\pi_i^*$ can be seen from Fig. 1(a). Suppose we like to find the MCDS of $V_4$ dominating $V_{4,7}$. If we consider the smaller number to the right of 7 in $\pi$, we can find 2. Note that vertex 2 dominates 3, 5 and 7. Thus, we can say that the MCDS of $V_4^*$ dominating $V_{4,7}$ is $\{3, 2\}$ which is significantly smaller. In some sense, we have obtained a smaller dominating set by looking ahead.

For any vertex set $S$, define $\max(S)$ to be the maximum number in $S$. For each $i$ and $j$, $1 \leqslant i, j \leqslant n$, define $D_{i,j}$ as follows:

1. $D_{i,j}$ is a minimum cardinality subset of $V_i^*$ dominating $V_{i,j}$.
2. $\max(D_{i,j})$ is as large as possible.

Obviously, $D_{n,n}$ is a desired minimum cardinality dominating set for $G$.

Let $X$ be a set of subsets of $V$. Define set_min($X$) as follows: set_min($X$) $= \phi$ if $X = \phi$ or $X = \{\phi\}$, and set_min($X$) $= A$ where $A$ is a non-empty element in $X$ such that $A$ is with the minimum cardinality among all elements in $X$ and $\max(A)$ is as large as possible, if otherwise. set_min($X$) may not be unique. If there are more than one candidate for set_min($X$), select any one to be set_min($X$). It is easy to prove that if $X, Y$ and $Z$ are sets of subsets of $V$ and $X = Y \cup Z$, we have $\text{set\_min}(X) = \text{set\_min}(\{\text{set\_min}(Y), \text{set\_min}(Z)\})$.

According to the definitions of $D_{i,j}$ and set_min, we have the following:

$$D_{i,j} = \begin{cases} \phi & \text{if } V_{i,j} = \phi, \\ \text{set\_min}(\{S \mid S \subset V_i^* \text{ and } S \rhd V_{i,j}\}) & \text{if otherwise.} \end{cases}$$

Consider the case where $i = 1$. If $j < \pi_1$, obviously, $V_{1,j} = \{\pi_1\} \cap \{1, 2, \ldots, j\} = \phi$. Hence, $D_{i,j} = \phi$. Otherwise, $V_{1,j} = \{\pi_1\}$. Since $\pi_1^* = 1$, $V_1^* = \{\pi_1, 1\}$. We have $D_{1,j} = \{\pi_1\}$. We therefore have the following rule:

$$D_{1,j} = \begin{cases} \phi & \text{if } j < \pi_1, \\ \{\pi_1\} & \text{if otherwise.} \end{cases}$$

For the case of $1 < i \leqslant n$, we prove the following lemmas before presenting the recursive formula of $D_{i,j}$.

**Lemma 1.** *For each $i_1, i_2$ and $j$, $1 \leqslant i_1 < i_2 \leqslant n$ and $1 \leqslant j \leqslant n$, $V_{i_1,j} \subset V_{i_2,j}$ and $V_{i_1}^* \subset V_{i_2}^*$.*

**Proof.** The proof is straightforward and omitted. □

For $1 < i \leqslant n$, let $D_{\pi_i^*} = D_{i-1,\pi_i^*} \cup \{\pi_i^*\}$, $D_{\pi_i} = D_{i-1,\pi_i} \cup \{\pi_i\}$ and $D_{\max} = D_{i-1,j} \cup \{\max(V_i)\}$. The following lemma claims that $D_{\pi_i^*}$, $D_{\pi_i}$ and $D_{\max}$ are candidates for computing $D_{i,j}$.

**Lemma 2.** *For each $i$ and $j$, $1 < i \leqslant n$ and $1 \leqslant j \leqslant n$,*

$$\{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\} \subset \{S \mid S \subset V_i^* \text{ and } S \rhd V_{i,j}\}.$$

**Proof.** Since, by definition, $D_{i-1,\pi_i^*} = \text{set\_min}(\{S \mid S \subset V_{i-1}^* \text{ and } S \rhd V_{i-1,\pi_i^*}\})$, we have $D_{i-1,\pi_i^*} \subset V_{i-1}^*$ and $D_{i-1,\pi_i^*} \rhd V_{i-1,\pi_i^*}$. According to Lemma 1, $V_{i-1}^* \subset V_i^*$. Therefore, we have $D_{i-1,\pi_i^*} \subset V_i^*$. Hence (1) $D_{i-1,\pi_i^*} \cup \{\pi_i^*\} \subset V_i^*$. Consider the set $V_{i,j} \backslash V_{i-1,\pi_i^*}$. Suppose that $x \in V_{i,j} \backslash V_{i-1,\pi_i^*}$, and $x \neq \pi_i^*$. It is easy to prove that $x > \pi_i^*$ and $\pi_x^{-1} < \pi_{\pi_i^*}^{-1}$. Hence $(x - \pi_i^*)(\pi_x^{-1} - \pi_{\pi_i^*}^{-1}) < 0$. That is, $(x, \pi_i^*) \in E$ and $\{\pi_i^*\} \rhd V_{i,j} \backslash V_{i-1,\pi_i^*}$. Note that $D_{i-1,\pi_i^*} \rhd V_{i-1,\pi_i^*}$. Hence (2) $V_{i,j}$ is dominated by $D_{i-1,\pi_i^*} \cup \{\pi_i^*\}$. According to (1) and (2), we have $D_{i-1,\pi_i^*} \cup \{\pi_i^*\} \in \{S \mid S \subset V_i^* \text{ and } S \rhd V_{i,j}\}$.

The proofs for $D_{i-1,\pi_i} \cup \{\pi_i\}$ and $D_{i-1,j} \cup \{\max(V_i)\}$ are similar and omitted. □

**Lemma 3.** *For each $i$ and $j$, $1 < i \leqslant n$ and $\pi_i^* \leqslant j \leqslant n$, if $Y \subset \{S \mid \pi_i^* \in S, S \subset V_i^*$ and $S \rhd V_{i,j}\}$, then $\text{set\_min}(Y \cup \{D_{\pi_i^*}\}) = D_{\pi_i^*}$.*

**Proof.** Let $X$ denote the set $\{S \mid \pi_i^* \in S, S \subset V_i^* \text{ and } S \rhd V_{i,j}\}$. Suppose $Y \subset X$. If $Y$ is an empty set, the proof is complete. Suppose $Y$ is not empty. Consider any vertex set $A \in Y$. Since $j \geqslant \pi_i^*$, by Lemma 1, we have $V_{i,j} \supset V_{i-1,\pi_i^*}$. Since $V_{i,j} \supset V_{i-1,\pi_i^*}$ and $A \rhd V_{i,j}$, $A \rhd V_{i-1,\pi_i^*}$. Note that no vertex in $V_{i-1,\pi_i^*}$ is dominated by $\pi_i$ or $\pi_i^*$. Hence,

$A \backslash \{\pi_i, \pi_i^*\} \rhd V_{i-1,\pi_i^*}$. Since $A \subset V_i^* = V_{i-1} \cup \{\pi_i, \pi_i^*\}$, $A \setminus \{\pi_i, \pi_i^*\} \subset V_{i-1} \subset V_{i-1}^*$. Therefore, $A \setminus \{\pi_i, \pi_i^*\} \in \{S \mid S \subset V_{i-1}^*$ and $S \rhd V_{i-1,\pi_i^*}\}$. Since $D_{i-1,\pi_i^*} = \text{set\_min}\{S \mid S \subset V_{i-1}^*$ and $S \rhd V_{i-1,\pi_i^*}\}$, we have $|D_{i-1,\pi_i^*}| \leqslant |A \setminus \{\pi_i, \pi_i^*\}|$ and if $|D_{i-1,\pi_i^*}| = |A \setminus \{\pi_i, \pi_i^*\}|$, max $(D_{i-1,\pi_i^*}) \geqslant \max(A \backslash \{\pi_i, \pi_i^*\})$. Note that $\pi_i^*$ must belong to $A$. Then, $|D_{i-1,\pi_i^*}| \leqslant |A| - 1$ and $|D_{i-1,\pi_i^*}| = |A| - 1$ if and only if $\pi_i^* = \pi_i$ or $\pi_i \notin A$. Hence, we have $|D_{i-1,\pi_i^*} \cup \{\pi_i^*\}| \leqslant |A|$ and if $|D_{i-1,\pi_i^*} \cup \{\pi_i^*\}| = |A|$, $\max(D_{i-1,\pi_i^*} \cup \{\pi_i^*\}) \geqslant \max(A)$. Therefore, set_min $(Y \cup \{D_{\pi_i^*}\}) = D_{\pi_i^*}$. □

**Lemma 4.** *For each i and j, $1 < i \leqslant n$ and $\pi_i \leqslant j \leqslant n$, if $Y \subset \{S \mid \pi_i \in S, \pi_i^* \notin S, S \subset V_i^*$ and $S \rhd V_{i,j}\}$, then set_min$(Y \cup \{D_{\pi_i}\}) = D_{\pi_i}$.*

**Proof.** Let $X$ denote the set $\{S \mid \pi_i \in S,\ \pi_i^* \notin S,\ S \subset V_i^*$ and $S \rhd V_{i,j}\}$. Suppose $Y \subset X$. If $Y$ is an empty set, the proof is complete. Suppose $Y$ is not empty. Consider any vertex set $A \in Y$. Since $j \geqslant \pi_i$, by Lemma 1, we have $V_{i,j} \supset V_{i-1,\pi_i}$. Since $V_{i,j} \supset V_{i-1,\pi_i}$ and $A \rhd V_{i,j}$, $A \rhd V_{i-1,\pi_i}$. Note that no vertex in $V_{i-1,\pi_i}$ is dominated by $\pi_i$. Hence, $A \backslash \{\pi_i\} \rhd V_{i-1,\pi_i}$. Since $A \subset V_i^*$ and $\pi_i^* \notin A$, $A \backslash \{\pi_i\} \subset V_{i-1} \subset V_{i-1}^*$. Therefore, $A \backslash \{\pi_i\} \in \{S \mid S \subset V_{i-1}^*$ and $S \rhd V_{i-1,\pi_i}\}$. Since $D_{i-1,\pi_i} = \text{set\_min}\{S \mid S \subset V_{i-1}^*$ and $S \rhd V_{i-1,\pi_i}\}$, we have $|D_{i-1,\pi_i}| \leqslant |A \backslash \{\pi_i\}|$ and if $|D_{i-1,\pi_i}| = |A \backslash \{\pi_i\}|$, $\max(D_{i-1,\pi_i}) \geqslant \max(A \backslash \{\pi_i\})$. Note that $\pi_i \in A$. Hence, we have $|D_{i-1,\pi_i} \cup \{\pi_i\}| \leqslant |A|$ and if $|D_{i-1,\pi_i} \cup \{\pi_i\}| = |A|$, max $(D_{i-1,\pi_i} \cup \{\pi_i\}) \geqslant \max(A)$. Therefore, set_min$(Y \cup \{D_{\pi_i}\}) = D_{\pi_i}$. □

**Lemma 5.** *For each i and j, $1 < i \leqslant n$ and $\pi_i \leqslant j \leqslant n$, if $\max(D_{i-1,j}) < \pi_i$ and $Y \subset \{S \mid \pi_i \notin S,\ \pi_i^* \notin S,\ S \subset V_i^*$ and $S \rhd V_{i,j}\}$, then set_min$(Y \cup \{D_{\max}\}) = D_{\max}$.*

**Proof.** Let $X$ denote the set $\{S \mid \pi_i \notin S,\ \pi_i^* \notin S,\ S \subset V_i^*$ and $S \rhd V_{i,j}\}$. Let $Z$ denote the set $\{S \mid S \subset V_{i-1}^*$ and $S \rhd V_{i-1,j}\}$. Suppose $\max(D_{i-1,j}) < \pi_i$ and $Y \subset X$. If $Y$ is an empty set, the proof is complete. Suppose $Y$ is not empty. Consider any vertex set $A \in Y$. Since $Y \subset X$, $A \in X$. Since $A \rhd V_{i,j}$ and $V_{i,j} \supset V_{i-1,j}$, $A \rhd V_{i-1,j}$. Since $\pi_i \notin A$ and $\pi_i^* \notin A$, $A \subset V_{i-1}^*$. Therefore, we have $A \in Z$. Since $\pi_i \leqslant j$ and $A \rhd V_{i,j}$, we have $A \rhd \{\pi_i\}$. Furthermore, since $\pi_i \notin A$, $\pi_i^* \notin A$ and $A \rhd \{\pi_i\}$, it can be proved that $\max(A) > \pi_i$. Note that $\max(D_{i-1,j}) < \pi_i$. Hence, $\max(A) > \max(D_{i-1,j})$. Note that $D_{i-1,j} = \text{set\_min}(Z)$. Since $A \in Z$ and $\max(A) > \max(D_{i-1,j})$, it must be true that $|A| > |D_{i-1,j}|$. Hence $|A| \geqslant D_{i-1,j}| + 1 = |D_{i-1,j} \cup \{\max(V_i)\}|$. Furthermore, since $A \subset V_i$, we have $\max(A) \leqslant \max(V_i) = \max(D_{i-1,j} \cup \{\max(V_i)\})$. Therefore, set_min$(Y \cup \{D_{\max}\}) = D_{\max}$. □

**Lemma 6.** *For each i and j, $1 < i \leqslant n$ and $1 \leqslant j \leqslant n$, if $Y \subset \{S \mid \pi_i \notin S,\ \pi_i^* \notin S,\ S \subset V_i^*$ and $S \rhd V_{i,j}\}$, set_min$(Y \cup \{D_{i-1,j}\}) = D_{i-1,j}$.*

**Proof.** Let $X$ denote the set $\{S \mid \pi_i \notin S,\ \pi_i^* \notin S,\ S \subset V_i^*$ and $S \rhd V_{i,j}\}$. Let $Z$ denote the set $\{S \mid S \subset V_{i-1}^*$ and $S \rhd V_{i-1,j}\}$. Suppose $Y \subset X$. If $Y$ is an empty set, the proof is complete. Suppose $Y$ is not empty. Consider any vertex set $A \in Y$. Since $\pi_i \notin A$, $\pi_i^* \notin A$ and $V_i^* = V_{i-1} \cup \{\pi_i^*, \pi_i\}$, we have $A \subset V_{i-1} \subset V_{i-1}^*$. Since, by Lemma 1, $V_{i-1,j} \subset V_{i,j}$ and $A \rhd V_{i,j}$, we have $A \rhd V_{i-1,j}$. Hence, $A \in Z$. Therefore, $Y \subset X \subset Z$. Note that $D_{i-1,j} =$

set_min($Z$). Therefore, we have set_min($Y \cup \{D_{i-1,j}\}$) = set_min($\{$set_min($Y$), set_min ($Z$)$\}$). Since $Y \subset Z$, set_min($\{$set_min($Y$), set_min($Z$)$\}$) = set_min($Z$) = $D_{i-1,j}$. □

In the following, we present the recursive formula of our dynamic programming.

**Theorem 1.** *The following recursive formula correctly computes $D_{i,j}$, where $1 < i \leqslant n$ and $1 \leqslant j \leqslant n$,*

$$D_{i,j} = \begin{cases} \mathit{set\_min}(\{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\}) & \text{if } j \geqslant \pi_i \text{ and } \max(D_{i-1,j}) < \pi_i, \\ \mathit{set\_min}(\{D_{i-1,j}, D_{\pi_i^*}, D_{\pi_i}\}) & \text{if otherwise.} \end{cases}$$

**Proof.** Let $X = \{S \mid S \subset V_i^* \text{ and } S \rhd V_{i,j}\}$, $X_1 = \{S \mid \pi_i^* \in S,\ S \subset V_i^* \text{ and } S \rhd V_{i,j}\}$, $X_2 = \{S \mid \pi_i \in S,\ \pi_i^* \notin S,\ S \subset V_i^* \text{ and } S \rhd V_{i,j}\}$ and $X_3 = \{S \mid \pi_i \notin S,\ \pi_i^* \notin S,\ S \subset V_i^* \text{ and } S \rhd V_{i,j}\}$. It is obvious that $X = X_1 \cup X_2 \cup X_3$. According to Lemma 2 $\{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\} \subset X$. Hence,

$$\begin{aligned} \text{set\_min}(X) &= \text{set\_min}(X \cup \{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\}) \\ &= \text{set\_min}(X_1 \cup X_2 \cup X_3 \cup \{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\}) \\ &= \text{set\_min}(\{\text{set\_min}(X_1 \cup \{D_{\pi_i^*}\}), \text{set\_min}(X_2 \cup \{D_{\pi_i}\}), \\ &\qquad \text{set\_min}(X_3 \cup \{D_{\max}\})\}). \end{aligned}$$

Furthermore, if $D_{i-1,j} \in X$, since $|D_{i-1,j}| < |D_{\max}|$, then

$$\begin{aligned} \text{set\_min}(X) &= \text{set\_min}(X \cup \{D_{\pi_i^*}, D_{\pi_i}, D_{\max}, D_{i-1,j}\}) \\ &= \text{set\_min}(X_1 \cup X_2 \cup X_3 \cup \{D_{\pi_i^*}, D_{\pi_i}, D_{i-1,j}\}) \\ &= \text{set\_min}(\{\text{set\_min}(X_1 \cup \{D_{\pi_i^*}\}), \text{set\_min}(X_2 \cup \{D_{\pi_i}\}), \\ &\qquad \text{set\_min}(X_3 \cup \{D_{i-1,j}\})\}). \end{aligned}$$

*Case* 1. $j \geqslant \pi_i$ and $\max(D_{i-1,j}) < \pi_i$:
According to Lemmas 3–5, we have

$$\begin{aligned} \text{set\_min}(X) &= \text{set\_min}(\{\text{set\_min}(X_1 \cup \{D_{\pi_i^*}\}), \text{set\_min}(X_2 \cup \{D_{\pi_i}\}), \\ &\qquad \text{set\_min}(X_3 \cup \{D_{\max}\})\}). \\ &= \text{set\_min}(\{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\}). \end{aligned}$$

*Case* 2. $j \geqslant \pi_i$ and $\max(D_{i-1,j}) \geqslant \pi_i$:
We first claim that $D_{i-1,j} \in X$. Let $k$ be the number such that $\pi_k = \max(D_{i-1,j})$. Since $1 \leqslant k \leqslant i$ and $\pi_k \geqslant \pi_i$, we have $(\pi_k - \pi_i)(k - i) \leqslant 0$. That is, $\pi_k = \pi_i$ or $(\pi_k, \pi_i) \in E$. Since $j \geqslant \pi_i$, $V_{i,j} = V_{i-1,j} \cup \{\pi_i\}$. Furthermore, since $\pi_k \in D_{i-1,j}$ and $D_{i-1,j} \rhd V_{i-1,j}$, we have $D_{i-1,j} \rhd V_{i,j}$. According to Lemma 1, we have $V_{i-1}^* \subset V_i^*$. Hence, $D_{i-1,j} \subset V_{i-1}^* \subset V_i^*$. Since $D_{i-1,j} \rhd V_{i,j}$ and $D_{i-1,j} \subset V_i^*$, we have $D_{i-1,j} \in X$. According to Lemmas 3,

4 and 6, we have:

$$\begin{aligned}\text{set\_min}(X) = \text{set\_min}(\{&\text{set\_min}(X_1 \cup \{D_{\pi_i^*}\}), \text{set\_min}(X_2 \cup \{D_{\pi_i}\}),\\ &\text{set\_min}(X_3 \cup \{D_{i-1,j}\})\}).\\ = \text{set\_min}(\{&D_{\pi_i^*}, D_{\pi_i}, D_{i-1,j}\}).\end{aligned}$$

*Case* 3. $\pi_i^* \leqslant j < \pi_i$:
Since $j < \pi_i$, $V_{i,j} = V_{i-1,j}$. Hence, $D_{i-1,j} \triangleright V_{i,j}$. Since, according to Lemma 1, $V_{i-1}^* \subset V_i^*$, we have $D_{i-1,j} \subset V_{i-1}^* \subset V_i^*$. Therefore, $D_{i-1,j} \in X$. Furthermore, since $j < \pi_i$, no vertex in $V_{i,j}$ is dominated by $\pi_i$. Hence $\pi_i$ must not belong to $D_{i,j}$. Otherwise we can delete $\pi_i$ from $D_{i,j}$ and the remaining set still dominates $V_{i,j}^*$. This is directly contradictory to the definition of $D_{i,j}$. Therefore, $X_2$ and $D_{\pi_i}$ need not be considered for $D_{i,j}$. Then, according to Lemmas 3 and 6,

$$\begin{aligned}\text{set\_min}(X) &= \text{set\_min}(\{\text{set\_min}(X_1 \cup \{D_{\pi_i^*}\}), \text{set\_min}(X_3 \cup \{D_{i-1,j}\})\})\\ &= \text{set\_min}(\{D_{\pi_i^*}, D_{i-1,j}\}).\end{aligned}$$

Note that $D_{\pi_i} \in X$. Hence, the following equality holds:

$$\text{set\_min}(X) = \text{set\_min}(\{D_{\pi_i^*}, D_{i-1,j}\}) = \text{set\_min}(\{D_{\pi_i^*}, D_{\pi_i}, D_{i-1,j}\}).$$

*Case* 4. $j < \pi_i^*$:
It can be easily shown that $D_{i-1,j} \in X$. Furthermore, since $j < \pi_i^* \leqslant \pi_i$, no vertex in $V_{i,j} = V_{i-1,j}$ is dominated by $\pi_i^*$ or $\pi_i$. Therefore, neither $\pi_i^*$ nor $\pi_i$ belongs to $D_{i,j}$. Hence, we need not consider $X_1, X_2, D_{\pi_i^*}$ and $D_{\pi_i}$ for finding $D_{i,j}$. According to Lemma 6, we have

$$\text{set\_min}(X) = \text{set\_min}(X_3 \cup \{D_{i-1,j}\}) = D_{i-1,j}.$$

Note that both $D_{\pi_i}$ and $D_{\pi_i^*}$ belong to $X$. Hence, the following equality holds

$$\text{set\_min}(X) = D_{i-1,j} = \text{set\_min}(\{D_{\pi_i^*}, D_{\pi_i}, D_{i-1,j}\}).$$

Note that all cases are considered. Hence, the proof is complete. □

## 3. The new updating rules

If we try to find the minimum cardinality dominating set by applying Theorem 1, the algorithm is at least of the order of $O(n^2)$, as there are $O(n^2)$ $D_{i,j}$'s to compute. In the following, we prove that for a specified $i$, there is a monotone ordering among all $D_{i,j}$'s, $1 \leqslant j \leqslant n$. By using this monotone ordering, we propose new updating rules and in the next section, we shall show that we can obtain an optimal linear time algorithm from them.

Table 1
The new updating rules for computing $(|D_{i,j}|, \max(D_{i,j}))$

| Conditions | $(\lvert D_{i,j}\rvert, \max(D_{i,j}))$ |
|---|---|
| *Case* (1) $\lvert D_{i-1,\pi_i}\rvert = \lvert D_{i-1,\pi_i^*}\rvert$ | |
| (1.1) $\lvert D_{i-1,j}\rvert = \lvert D_{i-1,\pi_i^*}\rvert + 1$ | |
| (1.1.1) $\max(D_{i-1,j}) < \pi_i$ | $(\lvert D_{i-1,j}\rvert, \pi_i)$ |
| (1.1.2) $\max(D_{i-1,j}) \geqslant \pi_i$ | $(\lvert D_{i-1,j}\rvert, \max(D_{i-1,j})$ |
| (1.2) $\lvert D_{i-1,j}\rvert = \lvert D_{i-1,\pi_i^*}\rvert$ | |
| (1.2.1) $\max(D_{i-1,j}) < \pi_i$ and $j \geqslant \pi$ | $(\lvert D_{i-1,j}\rvert + 1, \max(V_i))$ |
| (1.2.2) $\max(D_{i-1,j}) \geqslant \pi_i$ or $j < \pi$ | $(\lvert D_{i-1,j}\rvert, \max(D_{i-1,j}))$ |
| (1.3) $\lvert D_{i-1,j}\rvert < \lvert D_{i-1,\pi_i^*}\rvert$ | $(\lvert D_{i-1,j}\rvert, \max(D_{i-1,j}))$ |
| *Case* (2) $\lvert D_{i-1,\pi_i}\rvert = \lvert D_{i-1,\pi_i^*}\rvert + 1$ | |
| (2.1) $\lvert D_{i-1,j}\rvert = \lvert D_{i-1,\pi_i^*}\rvert + 1$ | |
| (2.2.1) $(\max(D_{i-1,j}) < \pi_i$ and $j \geqslant \pi)$ or $\max(D_{i-1,j}) < \pi_i^*$ | $(\lvert D_{i-1,j}\rvert, \max(D_{\pi_i^*}))$ |
| (2.2.2) $(\max(D_{i-1,j}) \geqslant \pi_i$ or $j < \pi)$ and $\max(D_{i-1,j}) \geqslant \pi_i^*$ | $(\lvert D_{i-1,j}\rvert, \max(D_{i-1,j}))$ |
| (2.2) $\lvert D_{i-1,j}\rvert \leqslant \lvert D_{i-1,\pi_i^*}\rvert$ | $(\lvert D_{i-1,j}\rvert, \max(D_{i-1,j}))$ |

**Lemma 7** (The Monotone Lemma). *For each i, if* $j_1 < j_2$, *then* $|D_{i,j_1}| \leqslant |D_{i,j_2}|$. *Furthermore, if* $j_1 < j_2$ *and* $|D_{i,j_1}| = |D_{i,j_2}|$, *then* $\max(D_{i,j_1}) \geqslant \max(D_{i,j_2})$.

**Proof.** Suppose $j_1 < j_2$. According to the definition of $V_{i,j}$, it can be obtained directly that $V_{i,j_1} \subset V_{i,j_2}$. Since $D_{i,j_2} \triangleright V_{i,j_2}$ and $V_{i,j_1} \subset V_{i,j_2}$, $D_{i,j_2} \triangleright V_{i,j_1}$. Note that $D_{i,j_2} \subset V_i^*$. Hence, $D_{i,j_2} \in \{S \mid S \subset V_i^* \text{ and } S \triangleright V_{i,j_1}\}$. Since $D_{i,j_1} = \text{set\_min}(\{S \mid S \subset V_i^* \text{ and } S \triangleright V_{i,j_1}\})$, we have $|D_{i,j_1}| \leqslant |D_{i,j_2}|$ and furthermore, if $|D_{i,j_1}| = |D_{i,j_2}|$, then $\max(D_{i,j_1}) \geqslant \max(D_{i,j_2})$. $\square$

**Lemma 8.** *For each i and j,* $1 < i \leqslant n$ *and* $1 \leqslant j \leqslant n$, $|D_{i-1,j}| \leqslant |D_{i-1,\pi_i^*}| + 1$. *Furthermore, if* $|D_{i-1,j}| = |D_{i-1,\pi_i^*}| + 1$, *then* $\max(D_{i-1,j}) \geqslant \max(D_{i-1,\pi_i^*})$.

**Proof.** Consider the set $V_{i-1,j} \setminus V_{i-1,\pi_i^*}$. If $V_{i-1,j} \setminus V_{i-1,\pi_i^*}$ is an empty set, we have $V_{i-1,j} \subset V_{i-1,\pi_i^*}$. Therefore, $D_{i-1,\pi_i^*}$ dominates $V_{i-1,j}$. Note that $D_{i-1,\pi_i^*} \subset V_{i-1}^*$ and $D_{i-1,j} = \text{set\_min}(\{S \mid S \subset V_{i-1}^* \text{ and } S \triangleright V_{i-1,j}\})$. Hence, we have $|D_{i-1,j}| \leqslant |D_{i-1,\pi_i^*}|$. Suppose $V_{i-1,j} \setminus V_{i-1,\pi_i^*}$ is not empty. For any vertex $v \in V_{i-1,j} \setminus V_{i-1,\pi_i^*}$, we have $\pi_i^* < v \leqslant j$ and $1 \leqslant \pi_v^{-1} \leqslant i-1$. Note that $\pi_{i-1}^* \leqslant \pi_i^*$ and $\pi^{-1}_{\pi^*_{i-1}} \geqslant i-1$. Hence, $\pi_{i-1}^* < v$ and $\pi^{-1}_{\pi^*_{i-1}} \geqslant \pi_v^{-1}$. We therefore have $(\pi_{i-1}^* - v)(\pi^{-1}_{\pi^*_{i-1}} - \pi_v^{-1}) < 0$. That is, $v$ is dominated by $\pi_{i-1}^*$. Hence, we have $D_{i-1,\pi_i^*} \cup \{\pi_{i-1}^*\} \triangleright V_{i-1,j}$. It is obvious that $D_{i-1,\pi_i^*} \cup \{\pi_{i-1}^*\} \subset V_{i-1}^*$. Therefore, $D_{i-1,\pi_i^*} \cup \{\pi_{i-1}^*\} \in \{S \mid S \subset V_{i-1}^* \text{ and } S \triangleright V_{i-1,j}\}$. Note that again $D_{i-1,j} = \text{set\_min}(\{S \mid S \subset V_{i-1}^* \text{ and } S \triangleright V_{i-1,j}\})$. Hence, $|D_{i-1,j}| \leqslant |D_{i-1,\pi_i^*} \cup \{\pi_{i-1}^*\}| = |D_{i-1,\pi_i^*}| + 1$. Furthermore, if $|D_{i-1,j}| = |D_{i-1,\pi_i^*}| + 1$, $\max(D_{i-1,j}) \geqslant \max(D_{i-1,\pi_i^*} \cup \{\pi_{i-1}^*\}) \geqslant \max(D_{i-1,\pi_i^*})$. $\square$

Our new updating rules are listed in Table 1.

**Theorem 2.** *The updating rules listed in Table* 1 *correctly compute* $|D_{i,j}|$ *and* max $(D_{i,j})$ *for* $1 < i \leqslant n$ *and* $1 \leqslant j \leqslant n$.

**Proof.** For each $i$ and $j$, $1 < i \leqslant n$ and $1 \leqslant j \leqslant n$, according to Lemma 8, we have $|D_{i-1,j}| \leqslant |D_{i-1,\pi_i^*}| + 1$. Obviously, for $j = \pi_i$, we have $|D_{i-1,\pi_i}| \leqslant |D_{i-1,\pi_i^*}| + 1$. Furthermore, since $\pi_i \geqslant \pi_i^*$, we have $|D_{i-1,\pi_i}| \geqslant |D_{i-1,\pi_i^*}|$ by Lemma 7. Hence, there are only two cases which should be considered: (1). $|D_{i-1,\pi_i}| = |D_{i-1,\pi_i^*}|$ and (2). $|D_{i-1,\pi_i}| = |D_{i-1,\pi_i^*}| + 1$.

*Case* (1) $|D_{i-1,\pi_i}| = |D_{i-1,\pi_i^*}|$:

Since $|D_{i-1,\pi_i}| = |D_{i-1,\pi_i^*}|$ and $\pi_i^* \leqslant \pi_i$, according to Lemma 7, we have $\max(D_{i-1,\pi_i^*}) \geqslant \max(D_{i-1,\pi_i})$.

(1.1) $|D_{i-1,j}| = |D_{i-1,\pi_i^*}| + 1$:

Since $|D_{i-1,j}| > |D_{i-1,\pi_i^*}| = |D_{i-1,\pi_i}|$, we have $j > \pi_i$ by Lemma 7.

(1.1.1) $\max(D_{i-1,j}) < \pi_i$:

According to Theorem 1, $D_{i,j} = \text{set\_min}(\{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\})$. Since, according to Lemma 8, $|D_{i-1,j}| = |D_{i-1,\pi_i^*}| + 1$, we have $\max(D_{i-1,j}) \geqslant \max(D_{i-1,\pi_i^*})$. Hence, $\pi_i > \max(D_{i-1,j}) \geqslant \max(D_{i-1,\pi_i^*}) \geqslant \max(D_{i-1,\pi_i})$. Since $\pi_i \geqslant \pi_i^*$ and $\pi_i > \max(D_{i-1,\pi_i^*}) \geqslant \max(D_{i-1,\pi_i})$, we have $\max(D_{\pi_i}) = \pi_i \geqslant \max(D_{\pi_i^*})$. Note that $|D_{\pi_i^*}| = |D_{\pi_i}| = |D_{i-1,\pi_i^*}| + 1$ and $|D_{\max}| = |D_{i-1,j}| + 1 = |D_{i-1,\pi_i^*}| + 2$. Therefore, $D_{i,j} = \text{set\_min}(\{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\}) = D_{\pi_i}$. That is, $(|D_{\pi_i}|, \max(D_{\pi_i})) = (|D_{i-1,j}|, \pi_i)$.

(1.1.2) $\max(D_{i-1,j}) \geqslant \pi_i$:

According to Theorem 1, $D_{i,j} = \text{set\_min}(\{D_{i-1,j}, D_{\pi_i^*}, D_{\pi_i}\})$. Note that $|D_{i-1,j}| = |D_{\pi_i^*}| = |D_{\pi_i}| = |D_{i-1,\pi_i^*}| + 1$. Furthermore, we have max $(D_{i-1,j}) \geqslant \pi_i \geqslant \pi_i^*$ and $\max(D_{i-1,\pi_i^*}) \geqslant \max(D_{i-1,\pi_i})$ and by Lemma 8, $\max(D_{i-1,j}) \geqslant (D_{i-1,\pi_i^*})$. Hence, $D_{i,j} = D_{i-1,j}$. That is, $(|D_{i,j}|, \max(D_{i,j})) = (|D_{i-1,j}|, \max(D_{i-1,j}))$.

(1.2) $|D_{i-1,j}| = |D_{i-1,\pi_i^*}|$:

(1.2.1) $\max(D_{i-1,j}) < \pi_i$ and $j \geqslant \pi_i$:

By Theorem 1, $D_{i,j} = \text{set\_min}(\{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\})$. Since, $|D_{i-1,\pi_i^*}| = |D_{i-1,\pi_i}| = |D_{i-1,j}|$, $|D_{\pi_i^*}| = |D_{\pi_i}| = |D_{\max}| = |D_{i-1,\pi_i^*}| + 1$. Note that max $(V_i)$ is the largest number of $V_i^*$. Hence, $D_{i,j} = \text{set\_min}(\{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\}) = D_{\max}$. That is, $(|D_{i,j}|, \max(D_{i,j})) = (|D_{i-1,j}| + 1, \max(V_i))$.

(1.2.2) $\max(D_{i-1,j}) \geqslant \pi_i$ and $j < \pi_i$:

According to Theorem 1, $D_{i,j} = \text{set\_min}(\{D_{i-1,j}, D_{\pi_i^*}, D_{\pi_i}\})$. Note that $|D_{i-1,j}| < |D_{\pi_i^*}| = |D_{\pi_i}|$. Hence, $D_{i,j} = \text{set\_min}(\{D_{i-1,j}, D_{\pi_i^*}, D_{\pi_i}\}) = D_{i-1,j}$. That is, $(|D_{i,j}|, \max(D_{i,j})) = (|D_{i-1,j}|, \max(D_{i-1,j}))$.

(1.3) $|D_{i-1,j}| < |D_{i-1,\pi_i^*}|$:

According to Lemma 7, we have $j < \pi_i$ since $|D_{i-1,j}| < |D_{i-1,\pi_i}|$. Hence, by Theorem 1, we have $D_{i,j} = \text{set\_min}(\{D_{i-1,j}, D_{\pi_i^*}, D_{\pi_i}\})$. Note that $|D_{i-1,j}| < |D_{i-1,\pi_i^*}| < |D_{\pi_i^*}| = |D_{\pi_i}|$. Hence, $D_{i,j} = \text{set\_min}(\{D_{i-1,j}, D_{\pi_i^*}, D_{\pi_i}\}) = D_{i-1,j}$. That is, $(|D_{i,j}|, \max(D_{i,j})) = (|D_{i-1,j}|, \max(D_{i-1,j}))$.

Table 2
The values of $(|D_{i,j}|, \max(D_{i,j}))$'s for the permutation graph in Fig. 1

| $j=$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $i=1$ | $(0,0)$ | $(0,0)$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(1,3)$ |
| $i=2$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(1,3)$ |
| $i=3$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(2,5)$ | $(2,5)$ | $(2,5)$ |
| $i=4$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(2,5)$ | $(2,5)$ | $(2,3)$ |
| $i=5$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(2,7)$ | $(2,5)$ | $(2,5)$ | $(2,4)$ |
| $i=6$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(2,7)$ | $(2,5)$ | $(2,5)$ | $(2,4)$ |
| $i=7$ | $(1,3)$ | $(1,3)$ | $(1,3)$ | $(2,7)$ | $(2,5)$ | $(2,5)$ | $(3,7)$ |

*Case* (2) $|D_{i-1,\pi_i}| = |D_{i-1,\pi_i^*}| + 1$: In this case, we have $|D_{\pi_i}| = |D_{i-1,\pi_i^*}| + 2$.

(2.1) $|D_{i-1,j}| = |D_{i-1,\pi_i^*}| + 1$:

(2.1.1) ($\max(D_{i-1,j}) < \pi_i$ and $j \geqslant \pi_i$) or $\max(D_{i-1,j}) < \pi_i^*$:
If $\max(D_{i-1,j}) < \pi_i$ and $j \geqslant \pi_i$, by Theorem 1, $D_{i,j} = \text{set\_min}(\{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\})$. Note that $|D_{\pi_i^*}| = |D_{i-1,\pi_i^*}| + 1 < |D_{\pi_i}| = |D_{\max}| = |D_{i-1,\pi_i^*}| + 2$. Hence we have $D_{i,j} = \text{set\_min}(\{D_{\pi_i^*}, D_{\pi_i}, D_{\max}\}) = D_{\pi_i^*}$. That is, $(|D_{i,j}|, \max(D_{i,j})) = (|D_{i-1,j}|, \max(D_{\pi_i^*}))$. Otherwise, by Theorem 1, $D_{i,j} = \text{set\_min}(\{D_{i-1,j}, D_{\pi_i^*}, D_{\pi_i}\})$. Note that $|D_{i-1,j}| = |D_{\pi_i^*}| = |D_{i-1,\pi_i^*}| + 1 < |D_{\pi_i}| = |D_{i-1,\pi_i^*}| + 2$. If $\max(D_{i-1,j}) < \pi_i^*$, we have $\max(D_{i-1,j}) < \max(D_{\pi_i^*})$. Hence, $D_{i,j} = D_{\pi_i^*}$. That is, $(|D_{i,j}|, \max(D_{i,j})) = (|D_{i-1,j}|, \max(D_{\pi_i^*}))$.

(2.1.2) ($\max(D_{i-1,j}) \geqslant \pi_i$ or $j < \pi_i$) and $\max(D_{i-1,j}) \geqslant \pi_i^*$:
According to Theorem 1, $D_{i,j} = \text{set\_min}(\{D_{i-1,j}, D_{\pi_i^*}, D_{\pi_i}\})$. Note that $|D_{i-1,j}| = |D_{\pi_i^*}| = |D_{i-1,\pi_i^*}| + 1 < |D_{\pi_i}| = |D_{i-1,\pi_i^*}| + 2$. Since $|D_{i-1,j}| = |D_{i-1,\pi_i^*}| + 1$, we have $\max(D_{i-1,j}) \geqslant \max(D_{i-1,\pi_i^*})$ by Lemma 8. Since $\max(D_{i-1,j}) \geqslant \pi_i^*$ and $\max(D_{i-1,j}) \geqslant \max(D_{i-1,\pi_i^*})$, we have $\max(D_{i-1,j}) \geqslant \max(D_{\pi_i^*})$. Hence, $D_{i,j} = D_{i-1,j}$. That is, $(|D_{i,j}|, \max(D_{i,j})) = (|D_{i-1,j}|, \max(D_{i-1,j}))$.

(2.2) $|D_{i-1,j}| \leqslant |D_{i-1,\pi_i^*}|$:
Since $|D_{i-1,j}| \leqslant |D_{i-1,\pi_i^*}|$, we have $|D_{i-1,j}| < |D_{i-1,\pi_i^*}| + 1 = |D_{i-1,\pi_i}|$. According to Lemma 7, we have $j < \pi_i$. Hence, by Theorem 1, $D_{i,j} = \text{set\_min}(\{D_{i-1,j}, D_{\pi_i^*}, D_{\pi_i}\})$. Note that $|D_{i-1,j}| < |D_{\pi_i^*}| = |D_{i-1,\pi_i^*}| + 1 < |D_{\pi_i}| = |D_{i-1,\pi_i^*}| + 2$. Hence, $D_{i,j} = D_{i-1,j}$. That is, $(|D_{i,j}|, \max(D_{i,j})) = (|D_{i-1,j}|, \max(D_{i-1,j}))$. □

Table 2 shows the entire table of $(|D_{i,j}|, \max(D_{i,j}))$, for the example in Fig. 1.

From Lemma 7, we have that if $j_1 \leqslant j \leqslant j_2$ and $(|D_{i,j_1}|, \max(D_{i,j_1})) = (|D_{i,j_2}|, \max(D_{i,j_2}))$, then $(|D_{i,j_1}|, \max(D_{i,j_1})) = (|D_{i,j}|, \max(D_{i,j})) = (|D_{i,j_2}|, \max(D_{i,j_2}))$. Thus we do not have to compute all $|D_{i,j}|$'s and $\max(D_{i,j})$'s. We only have to compute the range of $j$'s such that $(|D_{i,j}|, \max(D_{i,j}))$ is the same in this range.

As shown in Table 3, we can redraw the above table in such a way that some entries are grouped into blocks. That is, for each $i$, divide the range of $j$ into longest blocks such that in each block, the 2-tuple $(|D_{i,j}|, \max(D_{i,j}))$ is the same for the range of $j$'s.

Table 3
The entries with the same $(|D_{i,j}|, \max(D_{i,j}))$ are grouped into blocks.

<table>
<tr><td>$j=$</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr>
<tr><td>$i=1$</td><td colspan="2">(0,0)</td><td colspan="5">(1,3)</td></tr>
<tr><td>$i=2$</td><td colspan="7">(1,3)</td></tr>
<tr><td>$i=3$</td><td colspan="4">(1,3)</td><td colspan="3">(2,5)</td></tr>
<tr><td>$i=4$</td><td colspan="4">(1,3)</td><td colspan="2">(2,5)</td><td>(2,3)</td></tr>
<tr><td>$i=5$</td><td colspan="3">(1,3)</td><td>(2,7)</td><td colspan="2">(2,5)</td><td>(2,4)</td></tr>
<tr><td>$i=6$</td><td colspan="3">(1,3)</td><td>(2,7)</td><td colspan="2">(2,5)</td><td>(2,4)</td></tr>
<tr><td>$i=7$</td><td colspan="3">(1,3)</td><td>(2,7)</td><td colspan="2">(2,5)</td><td>(3,7)</td></tr>
</table>

A block is called a $(d,m)$-block while $|D_{i,j}| = d$ and $\max(D_{i,j}) = m$ for any $j$ in its range. For instance, for $i = 1$, the range of $j = 3$ to 7 is now a (1,3)-block.

## 4. The linear time algorithm

Our final goal is to find all blocks and then trace back from the last block.

Let us first group all blocks with the same $|D_{i,j}| = d$ for any $j$ in their range into a $d$-group for each $i$. For example in Table 3, for $i = 5$, the blocks (2,7), (2,5) and (2,4) are now grouped into a 2-group. Similarly, blocks (2,7) and (2,5) are grouped into a 2-group for $i = 7$.

Consider Case (1) in Table 1. Note that when $|D_{i-1,j}| < |D_{i-1,\pi_i^*}|$, there is no updating as depicted in Case (1.3). That is, we only have to consider two $d$-groups, namely $|D_{i-1,\pi_i^*}|$-group and $(|D_{i-1,\pi_i^*}| + 1)$-group as depicted in Cases (1.1) and (1.2). As for Case 2, it is easy to see that we only have to consider the $(|D_{i-1,\pi_i^*}| + 1)$-group. We illustrate the situation in Fig. 2.

We now informally illustrate our algorithm. Suppose that we are in Case (1). We examine the $(|D_{i-1,\pi_i^*}|+1)$-group first. According to Lemma 7, inside every group, $\max(D_{i,j})$'s are not increasing as $j$ increases. Therefore in this group, we start from the rightmost block and move to the left one by one, until we hit a $(d,m)$-block where $m$ is greater than $\pi_i$. Before we hit this block, we apply the rule of Case (1.1.1). After that, we apply the rule of Case (1.1.2), as illustrated clearly in Fig. 2(a).

As can be seen in Fig. 2, throughout the entire algorithm, there are three mechanisms:

1. Several blocks combined into one block (Case (1.1.1), Case (1.2.1) and Case (2.1.1)).
2. A block split into two blocks (Case (1.2.1) and Case (2.1.1)).
3. No change (Case (1.1.2), Case (1.2.2), Case (1.3), Case (2.1.2) and Case (2.2)).

The following is our algorithm to solve the problem. We shall discuss the data structure needed to implement the algorithm later.

Algorithm A. Finding an MCDS on a Permutation Graph.
Input: A permutation $\pi = [\pi_1, \pi_2, \ldots, \pi_n]$.
Output: A minimum cardinality dominating set of the graph $G[\pi]$.

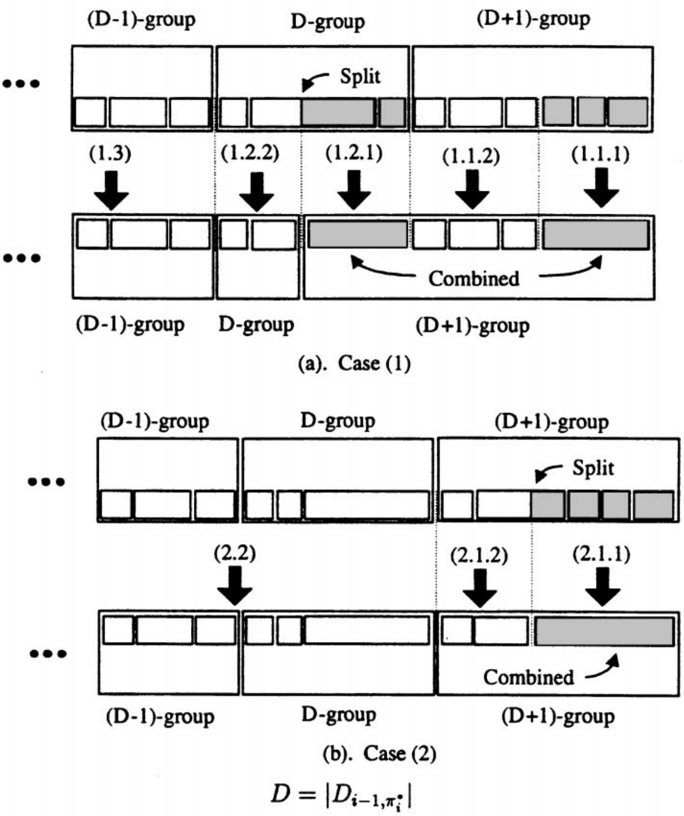


Fig. 2. The blocks updating by rules in Table 1.

*Step* 1: Compute $\pi_i^*$ and $\max(V_i)$ for all $i$'s.

*Step* 2: /$*$ initialization for $i = 1$ $*$/

Create (0,0)-block with the range $[1, \pi_1 - 1]$ and $(1, \pi_1)$-block with range $[\pi_1, n]$. Insert the (0,0)-block and the $(1, \pi_1)$-block into the 0-group and the 1-group, respectively.

*Step* 3: **for** $i \leftarrow 2$ **to** $n$ **do**

*Step* 3.1: /$*$ finding $|D_{i-1,\pi_i^*}|$ and $\max(D_{\pi_i^*})$ $*$/

Start from the block with range containing $\pi_{i-1}^*$ and move to the right block until we reach the $(d, m)$-block with range containing $\pi_i^*$. Then $|D_{i-1,\pi_i^*}| \leftarrow d$ and $\max(D_{\pi^*}) \leftarrow \max(\{m, \pi_i^*\})$.

*Step* 3.2: /$*$ finding $|D_{i-1,\pi_i}|$ $*$/

**if** the range of $|D_{i-1,\pi_i^*}|$-group contains $\pi_i$ **then** $|D_{i-1,\pi_i}| \leftarrow |D_{i-1,\pi_i^*}|$ **else** $|D_{i-1,\pi_i}| \leftarrow |D_{i-1,\pi_i^*}| + 1$.

*Step* 3.3: /$*$ updating data structures from $i - 1$ to $i$ $*$/

**case** $|D_{i-1,\pi_i}| = |D_{i-1,\pi_i^*}|$ **do** /$*$ Case (1) in Table 1 $*$/

*Step* 3.3.1: /* Case (1.1.1) in Table 1 */
Inside the $|D_{i-1,\pi_i^*}+1|$-group, find all $(d,m)$-blocks where $m<\pi_i$ by examining every block from the rightmost one to the left until a $(d,m)$-block is reached where $m \geqslant \pi_i$. Combine all found blocks into the $(d,\pi_i)$-block in this group.

*Step* 3.3.2: /* Case (1.2.1) in Table 1 */
Inside the $|D_{i-1,\pi_i^*}|$-group, find all $(d,m)$-blocks where $m<\pi_i$ and with range exceeding $\pi_i$ by examining every block from the rightmost one to the left until a $(d,m)$-block is reached where $m \geqslant \pi_i$ or the range of this block contains $\pi_i$. If the last block examined above is with $m<\pi_i$ and with range $[j_1,j_2]$ containing $\pi_i$ then split this block into two blocks with range $[j_1,\pi_i-1]$ and $[\pi_i,j_2]$. Combine all found blocks (including the block with range $[\pi_i,j_2]$) into the $(|D_{i-1,\pi_i^*}|+1, \max(V_i))$-block and insert this block into the $(|D_{i-1,\pi_i^*}|+1)$-group.

**case** $|D_{i-1,\pi_i}| = |D_{i-1,\pi_i^*}|+1$ **do** /* Case (2) in Table 1 */

*Step* 3.3.3: /* Case (2.1.1) in Table 1 */
Inside the $(|D_{i-1,\pi_i^*}|+1)$-group, find all $(d,m)$-blocks where $(m<\pi_i^*)$ or ($m<\pi_i$ and its range exceeding $\pi_i$) by examining every block from the rightmost one to the left until a $(d,m)$-block is reached where $(m \geqslant \pi_i^*)$ and ($m \geqslant \pi_i$ or the range of this block contains $\pi_i$). If the last $(d,m)$-block examined is with $\pi_i^* \leqslant m < \pi_i$ and with range $[j_1,j_2]$ containing $\pi_i$, then split this block into two blocks with range $[j_1,\pi_i-1]$ and $[\pi_i,j_2]$. Combine all the found blocks (including the block with range $[\pi_i,j_2]$), into the $(|D_{i-1,\pi_i}|, \max(D_{\pi_i^*}))$-block inside this group.

*Step* 4: Find am MCDS by backtracking from the last block produced.

In the following, we shall discuss the data structure needed to implement the above algorithm.

Within each $d$-group, there are several blocks. All of the blocks inside a group are linked by a doubly-linked list. Therefore, the locating of the leftmost and the rightmost blocks inside one group can be done in O(1) time. Our algorithm uses an additional pointer which points to the block with range containing $\pi_i^*$ for each iteration in Step 3. This pointer is used implicitly in Step 3.1 and may be updated while the blocks are split or combined in Step 3.3 in Algorithm A.

For time-complexity analysis, note the following facts:

1. In Step 1, the computation of all $\pi_i^*$'s can be done in $O(n)$ time by scanning the permutation array from right to left. Similarly, the computation of all $\max(V_i)$'s can be done in $O(n)$ time by scanning the permutation array from left to right.
2. Step 2 can be done in O(1) time.
3. In Step 3.1, because we always start from the previous block with range containing $\pi_{i-1}^*$ and transverse the linked lists to the right until we reach the block with range containing $\pi_i^*$, the number of blocks traversed is at most $(\pi_i^* - \pi_{i-1}^* + 1)$ in each

iteration $i$. The total number of blocks traversed must be smaller than $\sum_{2\leqslant i\leqslant n}(\pi_i^* - \pi_{i-1}^* + 1) = \pi_n^* - \pi_1^* + n - 1 < 2n$. Thus the total time needed to execute Step 3.1 is O($n$) time in amortized sense.
4. There are $n-1$ iterations in Step 3. Each iteration of Step 3.2 can be done in O(1) time. Therefore, the total execution time for Step 3.2 is O($n$).
5. All of the blocks examined in Steps 3.3.1–3.3.3 except the last block are combined into one block. Suppose there are $t_i$ blocks examined in Step 3.3 in each iteration $i$. In each iteration, our algorithm either goes through Steps 3.3.1 and 3.3.2 or through Step 3.3.3. If it goes through Steps 3.3.1 and 3.3.2, $t_i - 2$ blocks are combined. If it goes through Step 3.3.3. $t_i - 1$ blocks are combined. That is, at least $t_i - 2$ blocks are combined into other blocks. Since each iteration in Step 3.3 can generate at most 3 new blocks as illustrated in Fig. 2, there are totally at most $3n$ blocks generated in the algorithm. Since the total number of blocks combined cannot be larger than the number of new blocks generated, we have $\sum_{2\leqslant i\leqslant n} t_i - 2 \leqslant 3n$. That is, $\sum_{2\leqslant i\leqslant n} t_i < 5n$ and the total execution time for Step 3.3 is O($n$) is amortized sense.
6. Since there are $n-1$ iterations, the time needed for backtracking from the last block and finding an MCDS in O($n$) time.

From the above discussion, we conclude that the time-complexity of Algorithm A is O($n$) in amortized sense.

## 5. Conclusions

In this paper, we presented an algorithm for finding a minimum cardinality set for a permutation graph $G$. This algorithm exploits some subtle properties of this problem and runs in O($n$) time in amortized sense. Thus it is optimal.

## 6. For further reading

The following reference is also of interest to the reader: [13].

## References

[1] K. Arvind, C. Pandu Rangan, Connected domination and Steiner set on weighted permutation graphs, Inform. Process. Lett. 41 (1992) 215–220.
[2] M.J. Atallah, S.R. Kosaraju, An efficient algorithm for maxdominance, with applications, Algorithmica 4 (1989) 221–236.
[3] A. Brandstädt, D. Kratsch, On domination problems for permutation and other graphs, Theoret. Comput. Sci. 54 (1987) 181–198.
[4] C.J. Colbourn, L.K. Stewart, Permutation graphs: Connected domination and Steiner trees, Discrete Math. 86 (1990) 179–189.
[5] D.G. Corneil, L.K. Stewart, Dominating sets in perfect graphs, Discrete Math. 86 (1990) 145–164.
[6] M. Farber, J.M. Keil, Domination in permutation graphs, J. Algorithms 6 (1985) 309–321.

[7] M.R. Garey, D.S. Johnson, Computers and Intractability, A Guide to the Theory of NP-Completeness, Freeman, San Francisco, CA, 1979.
[8] M.C. Golumbic, Algorithmic Graph Theory and Perfect Graphs, Academic Press, New York, 1980.
[9] O.H. Ibarra, Q. Zheng, Some efficient algorithms for permutation graphs, J. Algorithms 16 (1994) 453–469.
[10] Y.D. Liang, Connected domination and Steiner trees in permutation graphs. In: ICS'94: 1994 Internat. Comput. Symp., 1994, pp. 13–18.
[11] Y.D. Liang, C. Rhee, S.K. Dhall, S. Lakshmivarahan, A new approach for the domination problem on permutation graphs, Inform. Process. Lett. 37 (1991) 219–224.
[12] C. Rhee, Y.D. Liang, S.K. Dhall, S. Lakshmivarahan, An $O(m + n)$ algorithm for finding minimum weight dominating set in permutation graphs, SIAM J. Comput. 25(2) (1996) 404–419.
[13] J. Spinrad, On comparability and permutation graphs, SIAM J. Comput. 14(3) (1985) 658–670.
[14] K.H. Tsai, W.L. Hsu, Fast algorithms for the dominating set problem on permutation graphs, Algorithmica 9 (1993) 601–614.